

## **PENGEMBANGAN APLIKASI GENERATOR MODUL PADA CMS UNTUK MEMPERMUDAH PENGEMBANGAN SISTEM BERBASIS WEB (studi kasus: CMS Lokomedia)**

Fiftin Noviyanto<sup>1</sup>, Al Mazari<sup>2</sup>  
Program Studi Teknik Informatika  
Universitas Ahmad Dahlan Yogyakarta  
<sup>1)</sup> [fiftin.noviyanto@tif.uad.ac.id](mailto:fiftin.noviyanto@tif.uad.ac.id)  
[almazary@gmail.com](mailto:almazary@gmail.com)

### **ABSTRAK**

CMS (*Content Management System*) merupakan salah satu aplikasi web yang memungkinkan penggunanya membuat website sendiri tanpa harus menguasai bahasa pemrograman. Lokomedia merupakan CMS buatan Indonesia yang saat ini banyak dikenal dan digunakan oleh masyarakat Indonesia untuk berbagai keperluan. Melalui buku – buku yang ditulis oleh pembuatnya, menjadikan CMS ini mudah untuk dipelajari dan dikembangkan oleh para pencintanya. Namun dalam perkembangannya, Lokomedia saat ini belum menggunakan sistem *plugin* seperti Wordpress dimana *website* yang akan dibangun terkoneksi dengan *plugin store* sehingga pengguna dengan mudah menambahkan *plugin* atau modul yang diinginkan tanpa melakukan konfigurasi, hal tersebut menuntut pengguna harus memahami arsitektur CMS dan mengerti bahasa pemrograman *web* untuk menambahkan modul baru.

Penelitian ini menggunakan model proses pengembangan sistem *Modified Waterfall* diawali dengan tahap analisis definisi persyaratan yaitu analisis kebutuhan dan merekayasa pengetahuan. Dilanjutkan dengan perancangan sistem dan perangkat lunak yaitu perancangan proses dan perancangan *interface*. Kemudian implementasi dan pengujian unit. Setelah itu sampai pada tahap terakhir yaitu integrasi pengujian sistem menggunakan metode *black box test* dan *beta test*. Aplikasi ini menggunakan PHP sebagai bahasa pemrograman.

Penelitian yang dilakukan menghasilkan aplikasi generator modul untuk mempermudah pengembangan sistem berbasis *web*. Aplikasi ini memiliki tiga langkah perancangan dan pada akhirnya menghasilkan kode program modul sesuai perancangan yang dilakukan, sehingga dapat mempercepat pembuatan modul dan meminimalkan kesalahan pengetikan kode program. Hasil uji coba menunjukkan bahwa aplikasi layak dan dapat digunakan.

**Kata Kunci :** *Generator Modul, Penghasil Modul, Penghasil Kode Program Modul, CMS Lokomedia.*

## 1. PENDAHULUAN

CMS (*Content Management System*) merupakan perangkat lunak yang dapat digunakan untuk membuat *website* dan manajemen konten *website* tersebut secara dinamis dan mudah. CMS Lokomedia yang dibuat oleh Lukmanul Hakim menggunakan PHP sebagai bahasa pemrogramannya dan Mysql sebagai tempat penyimpanan datanya merupakan salah satu CMS buatan Indonesia yang saat ini dikenal dan banyak digunakan oleh masyarakat Indonesia untuk berbagai keperluan. CMS Lokomedia memiliki dua fungsi standar yaitu untuk portal berita dan situs penjualan dapat didapatkan secara gratis melalui *website* resminya <http://bukulokomedia.com>. Melalui buku-buku yang ditulis oleh pembuatnya yang membahas tentang CMS Lokomedia, menjadikan CMS Lokomedia mudah untuk dipelajari dan dikembangkan oleh para pencintanya.

CMS Lokomedia saat ini belum menggunakan sistem *plugin* seperti CMS Wordpress dimana *website* yang dibuat terhubung dengan *plugin store* sehingga pengguna dapat dengan mudah menambahkan modul atau *plugin* yang dibutuhkan. Hal tersebut dikarenakan Lokomedia belum menyediakan wadah resmi bagi para pengembangnya untuk berkumpul menjadi satu dan menyajikan hasil karya mereka berupa modul - modul yang penting yang belum tersedia pada standarnya. Selain itu faktor lainnya adalah pembuat atau pengembang belum memperbaharui arsitektur Lokomedia yang memungkinkan sistem *plugin* diterapkan. Permasalahan tersebut menuntut para pengguna Lokomedia harus memahami arsitektur Lokomedia dan mengerti bahasa pemrograman *web* seperti PHP dan Mysql untuk menambahkan modul baru.

Modul - modul pada CMS Lokomedia memiliki arsitektur yang sama, perbedaannya hanya terletak pada jenis-jenis *field* dalam sebuah *form* pada suatu modul, artinya yang berbeda hanyalah data yang akan dimasukkan, dibaca, diubah dan dihapus, sedangkan proses insert, create, update dan delete adalah sama. Namun waktu yang diperlukan untuk menyelesaikan sebuah modul membutuhkan waktu yang cukup lama dengan konsentrasi penulisan kode program yang cukup tinggi dan dengan memperhatikan sisi keamanannya.

CMS Lokomedia saat ini juga belum menyediakan *tool* atau alat yang mempermudah dan mempercepat bagi para pengembang dalam merekayasa modul. Sehingga muncullah ide untuk membuat sebuah *tool* yang dapat mempermudah pembuatan modul dengan meminimalkan waktu dan kesalahan penulisan kode program tanpa mengurangi sisi keamanannya.

Latar belakang masalah di atas merupakan pengambilan topik penelitian ini. Judul penelitian ini adalah “ **Pengembangan Aplikasi Generator Modul Pada CMS Lokomedia Untuk Mempermudah Pengembangan Sistem Berbasis Web** ”.

## 2. KAJIAN PUSTAKA

Penelitian terdahulu yang pernah dilakukan oleh Fumio Narisawa, Hidemitsu Naya, Takanori Yokoyama pada tahun 1998 dengan judul “ **A Code Generator With Application-Oriented Size Optimization For Object-Oriented Embedded Control Software** ”. Penelitian tersebut merupakan pembuatan aplikasi generator

kode otomatis yang menghasilkan kode bahasa C dari object-oriented diagram spesifikasi [9]. Penelitian lainnya yang pernah dilakukan oleh Thorsten Sturm, Jesco von Voss, Marko Boger pada tahun 2002 dengan judul **“Generating Code From Uml With Velocity Templates”**. Penelitian tersebut merupakan pengembangan perangkat lunak penghasil kode otomatis berdasarkan template yang aktif mengakses informasi UML [10].

## 2.1. CMS (*Content Management System*)

CMS (*Content Management System*) adalah aplikasi *web* yang berisikan *template* untuk mengelola isi halaman *web* secara mudah [7]. Penggunaan *Content Management System* tidak memerlukan pengetahuan pemrograman *web* yang handal karena proses instalasi dan cara penggunaannya sudah *user friendly*. CMS sendiri ada yang dibuat khusus menyesuaikan kasus yang ada dan biasanya berbayar dan ada yang berupa *template* instan yang fungsionalitasnya dibuat dengan menyesuaikan pada beberapa proses bisnis yang ada di dunia nyata yang dapat digunakan secara gratis.

Aplikasi *Content Management System* instant yang banyak terdapat di *internet* saat ini kebanyakan dibuat menggunakan *scripting language* PHP dan database-nya adalah MySQL. Saat ini perkembangan *Content Management System* cukup pesat, banyak vendor yang membuat CMS instant yang didistribusikan secara gratis. Perkembangan CMS instant ini juga dipicu oleh perkembangan *web 2.0* yang memungkinkan interaksi dalam arti yang cukup luas antara pengelola *web* dan pengunjung *web*.

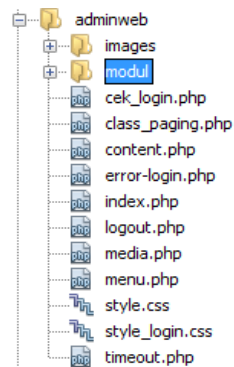
Selain perkembangan teknologi *web* dan infrastruktur *internet*, perkembangan pesat *Content Management System* juga dipicu oleh kebutuhan masyarakat dan pelaku bisnis yang menginginkan *web* dapat mendukung kegiatan bisnis mereka secara mudah dalam hal pengelolaan *content*, cepat dalam pembuatan *web*, serta murah dalam pengadaannya.

## 2.2. CMS Lokomedia

Lokomedia merupakan CMS yang dibuat oleh seorang praktisi *web* sekaligus penulis buku bertema komputer yaitu Lukmanul Hakim. Lokomedia pertama kali dibuat pada tahun 2008 yang sebelumnya merupakan *source code* yang diberikan pada buku **“Membongkar Trik Rahasia Para Master PHP”** [8].

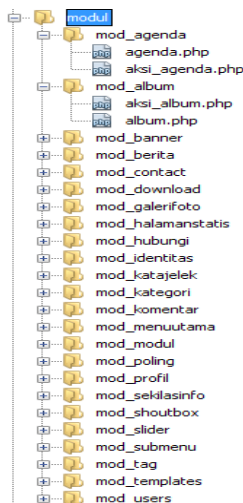
Lokomedia didistribusikan secara gratis pada *website* resminya <http://bukulokomedia.com> dan mempunyai forum resmi yaitu <http://forum.bukulokomedia.com>. Lokomedia banyak dikenal dan digunakan dikarenakan mudah digunakan dan dipelajari bagi para pemula yang belum mengenal OOP (*Object Oriented Programming*). Lokomedia juga selalu memberikan update versi saat buku baru diterbitkan oleh penulis sendiri yang membahas seputar teknologi *website* terbaru yang diterapkan pada versi terbarunya. Hal tersebut tentunya semakin memudahkan para pengguna untuk semakin mengetahui Lokomedia.

Arsitektur modul pada Lokomedia terlihat pada gambar berikut :



**Gambar 1.** Letak *Folder* Penyimpanan Modul

Gambar 1 menunjukkan letak penyimpanan *folder* modul. *Folder* modul disimpan didalam *folder* bernama modul yang terletak di dalam *folder* adminweb.



**Gambar 2.** Isi *Folder* Modul

Gambar 2 menunjukkan didalam *folder* bernama modul terdapat *folder-folder* modul itu sendiri, setiap modul diberinama depan mod\_ kemudian diikuti nama modul. Setiap modul hanya memiliki dua buah *file* yaitu *file* utamanya dan *file* sebagai aksinya, setiap modul juga memiliki nama yang unik yang digunakan untuk proses pemanggilan modul.

### 3. METODE PENELITIAN

Penelitian ini menggunakan model proses pengembangan sistem *Modified Waterfall* diawali dengan tahap analisis definisi persyaratan yaitu analisis kebutuhan. Dilanjutkan dengan perancangan sistem dan perangkat lunak yaitu perancangan proses dan perancangan *interface*. Kemudian implementasi dan pengujian unit. Setelah itu sampai pada tahap terakhir yaitu integrasi pengujian sistem menggunakan metode *black box test* dan *beta test*. Aplikasi ini menggunakan PHP sebagai bahasa pemrograman.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Analisis dan Definisi Persyaratan

#### a. Analisis Kebutuhan

Analisis kebutuhan data masukan yang diperlukan pada setiap *block* berupa:

- 1) Kebutuhan *input block setting* awal berupa: nama *database*, nama tabel jika sudah mempunyai tabel, lokasi penyimpanan *folder* modul, nama modul, nama *folder* modul, nama *file* utama, nama *file* aksi.
- 2) Kebutuhan *input block type form* berupa: tabel dari *database*, *type form* dari setiap *field*.
- 3) Kebutuhan *input block form*, pada *block* ini kebutuhan *input* setiap *type form* berbeda – beda terlihat pada tabel 1 berikut:

**Tabel 1.** Tabel Perbandingan Kebutuhan *Input Type Form*

Type Form	Kebutuhan Input			
	Label	Properti	Value	Validasi
Textfield	✓	✓	✓	✓
Hidden			✓	✓
Password	✓	✓	✓	✓
Textarea	✓	✓	✓	✓
Radio Button	✓		✓	✓
Radio Button Database	✓		✓	✓
Checkbox	✓		✓	✓
Checkbox Database	✓		✓	✓
Combobox	✓		✓	✓
Combobox Database	✓		✓	✓
File	✓		✓	✓
Date	✓		✓	✓

Analisis kebutuhan proses yang diperlukan pada setiap *block* berupa:

- 1) Proses *block setting* awal: koneksi *database*, pengambilan nama *database*, pengambilan nama - nama tabel, pengecekan lokasi penyimpanan *folder* modul, pengecekan informasi modul.
- 2) Proses *block type form*: menampilkan *field – field* tabel, proses pemilihan *type form*.
- 3) Proses *block form*: pemilihan halaman pengaturan, pengambilan *field – field* pada perintah SQL yang dimasukan.
- 4) Proses *block hasil generate*: pemilihan kode yang akan degenerate untuk setiap *type form*.

Analisis data keluaran difokuskan pada kode yang dihasilkan untuk tiap – tiap *type field* terlihat pada tabel 2 berikut :

**Tabel 2.** Tabel kode yang dihasilkan untuk *type field form*

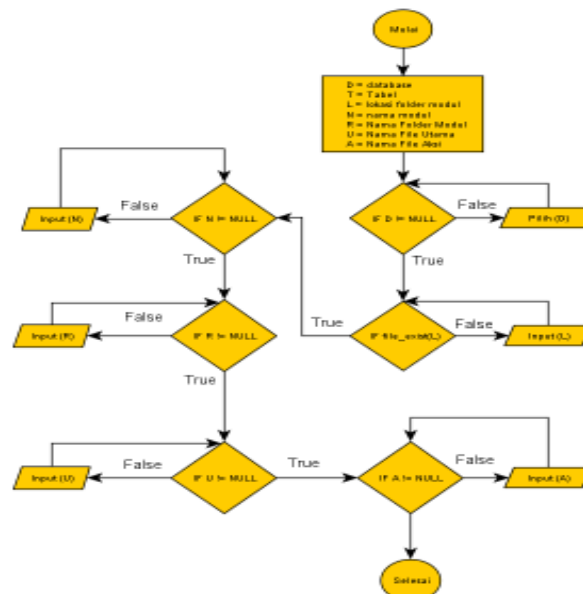
Type Form	Kode Hasil
Textfield	<input type='text' name="" size="" class="" style="" value="">
Hidden	<input type='hidden' name="" value="">
Password	<input type='password' name="" size="" class="" style="" value="">
Textarea	<textarea name="" class="" style="" rows="" cols="">value</textarea>
Radio Button	<input type='radio' name="" value=""> Label
Radio Button Database	<input type='radio' name="" value=""> Label 1 <input type='radio' name="" value=""> Label 2
Checkbox	<input type='checkbox' name="" value=""> Label
Checkbox Database	<input type='checkbox' name="" value=""> Label 1 <input type='checkbox' name="" value=""> Label 2
Combobox	<select name=""> <option value="">Label</option> </select>
Combobox Database	<select name=""> <option value="">Label 1</option> <option value="">Label 2</option> </select>
File	<input type='file' name='fupload' size="" class="" style="" value="">
Date	combotgl(1,31,'tgl_mulai',\$tgl_skrng); combonamabln(1,12,'bln_mulai',\$bln_sekarang); combothn(2000,\$thn_sekarang,'thn_mulai',\$thn_sekarang);

## 4.2. Perancangan Sistem dan Perangkat lunak

### a. Perancangan Proses

Di dalam perancangan proses digambarkan dalam bentuk flowchart. Berikut tahap-tahap perancangan proses :

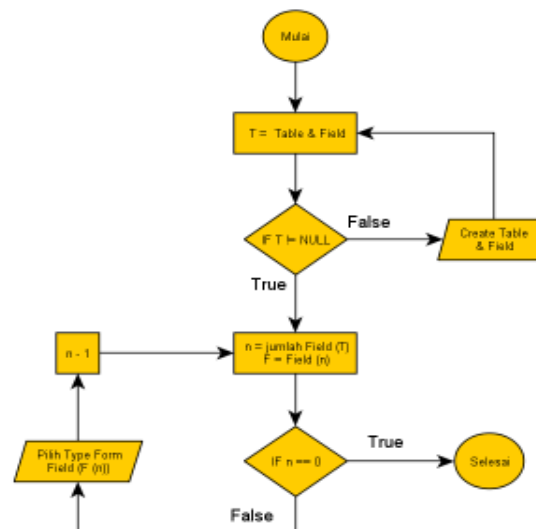
#### 1) Alur proses *block setting* awal



**Gambar 3.** Proses pengecekan seluruh informasi modul

Flowchart pada gambar 3 merupakan alur proses pengecekan keseluruhan data informasi modul yang akan di buat, jika semua informasi yang dibutuhkan telah terpenuhi, maka proses pada *block setting* awal telah selesai.

2) Alur proses *block type form*

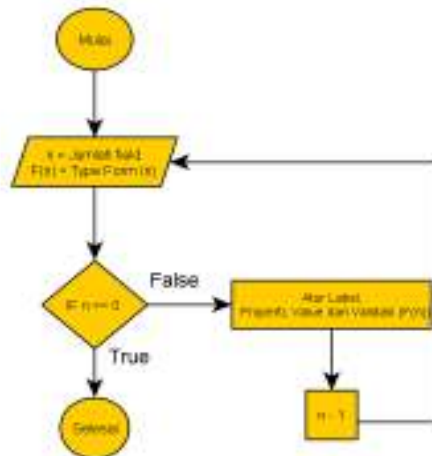


**Gambar 4.** Proses pada *block type form*

Flowchart pada Gambar 4 menunjukkan seluruh proses yang terjadi pada *block type form*, dimana proses diawali dengan melakukan cek apakah pengguna telah memiliki tabel dan *field* sebagai tempat penyimpanan data modul, jika belum maka sistem akan mengarahkan agar pengguna membuat tabel beserta *field*-nya. Kemudian setelah *field* ada, sistem akan menghitung berapa jumlah *field* dan mengetahui properti tiap-tiap *field*-nya. Kemudian sistem akan melakukan perulangan untuk menampilkan *field – field* yang

ada agar pengguna dapat memilih *type form* yang akan digunakan untuk tiap – tiap *field*.

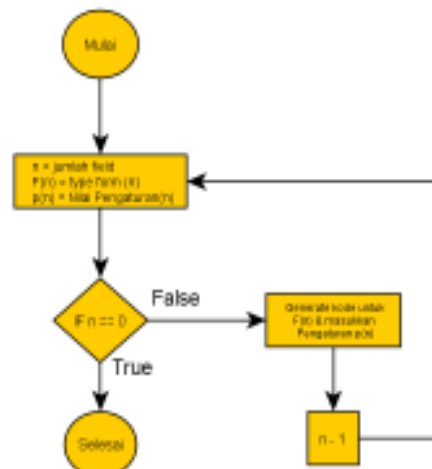
3) Alur proses *block form*



**Gambar 5.** Proses pada *block form*

Flowchart Gambar 5 menunjukkan proses yang terjadi pada *block form*, dimana proses diawali dengan mengetahui jumlah *field* dan *type form* yang digunakan untuk setiap *field* tersebut, setelah itu dilakukan perulangan untuk dapat menampilkan jenis – jenis pengaturan yang sesuai dengan *type form* pada tiap – tiap *field*nya.

4) Alur proses *block* hasil *generate*



**Gambar 6.** Proses pada *block* hasil *generate*

Flowchart pada Gambar 6 menunjukkan proses yang terjadi pada *block* hasil *generate*, dimana proses diawali dengan mengetahui jumlah *field* yang digunakan, *type form* untuk tiap – tiap *field* dan nilai pengaturan yang telah dilakukan pada tiap – tiap *field*. Setelah itu kode di generate sesuai *type form* dengan memasukkan pengaturan yang dilakukan.



## b. Perancangan *Interface*

Perancangan *interface* diperlukan untuk memberikan tampilan yang menarik dan memberikan kemudahan pengguna aplikasi. Perancangan *interface* terdiri dari perancangan menu dan perancangan *interface*.

### 4.1. Implementasi

Untuk mendapatkan kode program modul, maka perlu dilakukan perancangan modul. Aplikasi Generator akan memberikan informasi yang harus di berikan ke aplikasi. Berikut merupakan data yang harus diberikan pada setiap *block*nya:

Gambar 7. Tampilan *block setting* awal

*Block setting* awal digunakan untuk mendapatkan beberapa informasi yang berkaitan dengan modul yang akan dibuat, jika informasi yang dibutuhkan aplikasi telah terpenuhi maka aplikasi memperbolehkan untuk menuju ke step selanjutnya.

Gambar 8. Tampilan *block type form*

Gambar 8 merupakan tampilan jika tabel pada *block setting* awal belum dipilih, maka sistem akan mengarahkan untuk manajemen tabel, dan jika sudah memilih tabel yang akan digunakan maka sistem akan menyarankan untuk memilih *type form* untuk setiap *field* seperti berikut :

Field	Type	Null	Key	Default	Extra	Type Form
id_user	varchar	NO	PK		auto increment	Tampilan
gambar	varchar(200)	NO				Tampilan 2
judul	varchar(200)	NO				Tampilan 3
isi	text	NO				Tampilan 4

Gambar 9. Tampilan *block type form*

Gambar 9 menunjukkan *type form* yang harus di pilih untuk setiap *field* yang akan digunakan. Setelah mengatur *type form* maka step selanjutnya adalah mengatur *label*, *property*, *value* dan validasinya:

LABEL	PROPERTY	VALUE	VALIDATION
gambar	Type Form: File, Nama Field: gambar, Type: varchar(200) Size: 20, Class: <input type="text"/> CSS Style: <input type="text"/>	Nama Folder: <input type="text"/> Nama Gambar: <input type="text"/> Resize Width 1: <input type="text"/> Resize Width 2: <input type="text"/>	Boleh Kosong: <input type="checkbox"/> YA <input type="checkbox"/> TIDAK
judul	Type Form: Text Field, Nama Field: judul, Type: varchar(255) Size: 20, Class: <input type="text"/> CSS Style: <input type="text"/>	Value: <input type="text"/> Dengan SQL: <input type="text"/> Method Lain: <input type="text"/>	required: <input type="text"/>
isi	Type Form: Text Field, Nama Field: isi, Type: text Size: 20, Class: <input type="text"/> CSS Style: <input type="text"/>	Value: <input type="text"/> Dengan SQL: <input type="text"/> Method Lain: <input type="text"/>	required: <input type="text"/>

Gambar 10. Tampilan *block form*

*Block form* seperti pada gambar 10 merupakan tempat pengaturan untuk *label*, *property*, *value* dan validasi untuk setiap *field – field* yang digunakan. Setelah itu, *block* selanjutnya adalah *block* hasil generate:



Gambar 11 merupakan tampilan untuk kode program modul yang dihasilkan oleh generator. Terdapat dua buah kode program untuk *file* utama dan *file aksi* yang telah dideskripsikan pada *setting* awal. Generator juga akan membuat *folder* dan *file – file* modul secara otomatis kedalam *folder* yang diarahkan pada *setting* awal.

Agar block – block program dapat benar – benar bekerja menghasilkan kode program, yang perlu dilakukan adalah data input yang telah dideskripsikan pada analisis kebutuhan input untuk tiap – tiap block telah terpenuhi, sehingga tiap block dapat saling terhubung dan bekerja menghasilkan kode program modul.

Aplikasi ini menggunakan dua jenis pengujian yaitu *black box test* yang melibatkan pembuat CMS Lokomedia yaitu Lukmanul Hakim, dan *beta test* yang dilakukan oleh para *programmer* atau pengembang Lokomedia, sehingga dapat diperoleh tanggapan dari pemakai tentang aplikasi tersebut.

- Dari penelitian telah dirancang dan dibangun aplikasi generator modul untuk CMS Lokomedia dengan tiga langkah perancangan yang dapat di jalankan pada *local web server*.
- Aplikasi yang telah dirancang dan dibangun dapat mempercepat dalam membuat modul baru dan meminimalkan kesalahan dalam pengetikan program.
- Berdasarkan hasil uji sistem *black box test*, nilai presentase untuk jawaban “YA” adalah 100 % dan hasil uji *beta test* “ Sangat setuju = 44,8%, “Setuju = 55,2 %, sehingga perangkat lunak layak untuk digunakan serta sesuai dengan kebutuhan pemakai *user*.

1. Arsyad, Firmansyah., 2008, *Rekayasa CMS Open Source untuk Pembuatan Sistem Informasi Lowongan Kerja Online*, Skripsi S1, Universitas Ahmad Dahlan, Yogyakarta.
2. Hakim, Lukmanul., 2006, *Trik Rahasia Master PHP Terbongkar Lagi*, Lokomedia, Yogyakarta

3. Hakim, Lukmanul., 2008, *Membongkar Trik Rahasia Para Master PHP*, Lokomedia, Yogyakarta
4. Hakim, Lukmanul., 2010, *Bikin Website Super Keren dengan PHP dan JQuery*, Lokomedia, Yogyakarta
7. Hidayatullah, Fikri., 2010, *Pengembangan Component dan Module CMS Open Source untuk Pembuatan Aplikasi Hadits dan Terjemah Berbasis Web*, Skripsi S1, Universitas Ahmad Dahlan, Yogyakarta.
8. <http://bukulokomedia.com/>
9. [http://link.springer.com/chapter/10.1007%2F3-540-49255-0\\_168#page-1](http://link.springer.com/chapter/10.1007%2F3-540-49255-0_168#page-1)
10. [http://link.springer.com/chapter/10.1007/3-540-45800-X\\_13](http://link.springer.com/chapter/10.1007/3-540-45800-X_13)
11. <http://www.grocerycrud.com/>
12. <http://www.yiiframework.com/doc/guide/1.1/id/quickstart.first-app>
13. Kadir, Abdul., 2003, *Dasar Pemrograman Web Dinamis Menggunakan PHP*, Andi, Yogyakarta
14. MySQL Group, 2006, MySQL Documentation : <http://dev.mysql.com/doc/>
15. Peranginangin, Kasiman., 2006, *Aplikasi Web dengan PHP dan MySQL*, Andi, Yogyakarta.
16. PHP Documentation Group, 2006, *PHP Manual* : <http://www.php.net/manual/>
17. Sulhan, Moh., 2007, *Pengembangan Aplikasi Berbasis Web dengan PHP dan ASP*, Gava Media, Yogyakarta
18. Tarmuji, Ali., 2009, *Diktat Kuliah Rekayasa Perangkat Lunak*, Universitas Ahmad Dahlan, Yogyakarta